Section 3.0: CSENet 2000 Application Suite

3.0 CSENET 2000 APPLICATION SUITE

The purpose of the OCSE Child Support Enforcement Network (CSENet) 2000 Application Suite is to facilitate and support automation and transmission of interstate child support information. To use the services of the CSENet suite, state users electronically initiate and respond to CSE case activities in other states for locating noncustodial parents, establishing paternity and support obligations, enforcing support orders and collection of monies, and gathering additional case information.

The transmission of this information is contained in standardized data transactions that provide a common basis for data exchange between state CSE systems. These standardized transactions are transmitted from state to state via the CSENet suite over the OCSE Network. (Refer to Section 2.0: *OCSE Network Architecture* for more information about the network.)

The CSENet 2000 Application Suite provides services and support for:

- accessing and delivering transactions;
- verifying data integrity;
- data analysis;
- state readiness testing;
- Exchange Agreements verification;
- Interstate Roster and Referral Guide (IRG) data; and
- Management Information (M/I) Reports.

The suite consists of five software applications and two services that work together in concert to provide child support information and processing. These applications and services as well as the software management techniques used by CSENet are described in this section.

3.1 CSENet 2000 Transaction Exchange Process

The Transaction Exchange Process (TEP) is the data transfer process used by state systems to exchange child support enforcement information with other states via the CSENet 2000 Application Suite. The suite uses common transaction formats for generating and processing child support transactions. (For detailed information on these formats, see Appendix C*: Data Block Record Layout*.)

The CSENet suite exchanges transaction files with specified locations on the state system. These locations, usually data set names, are identified by states and maintained in the State Profile by the CSENet team. (For more information on the State Profile, see Appendix I.)

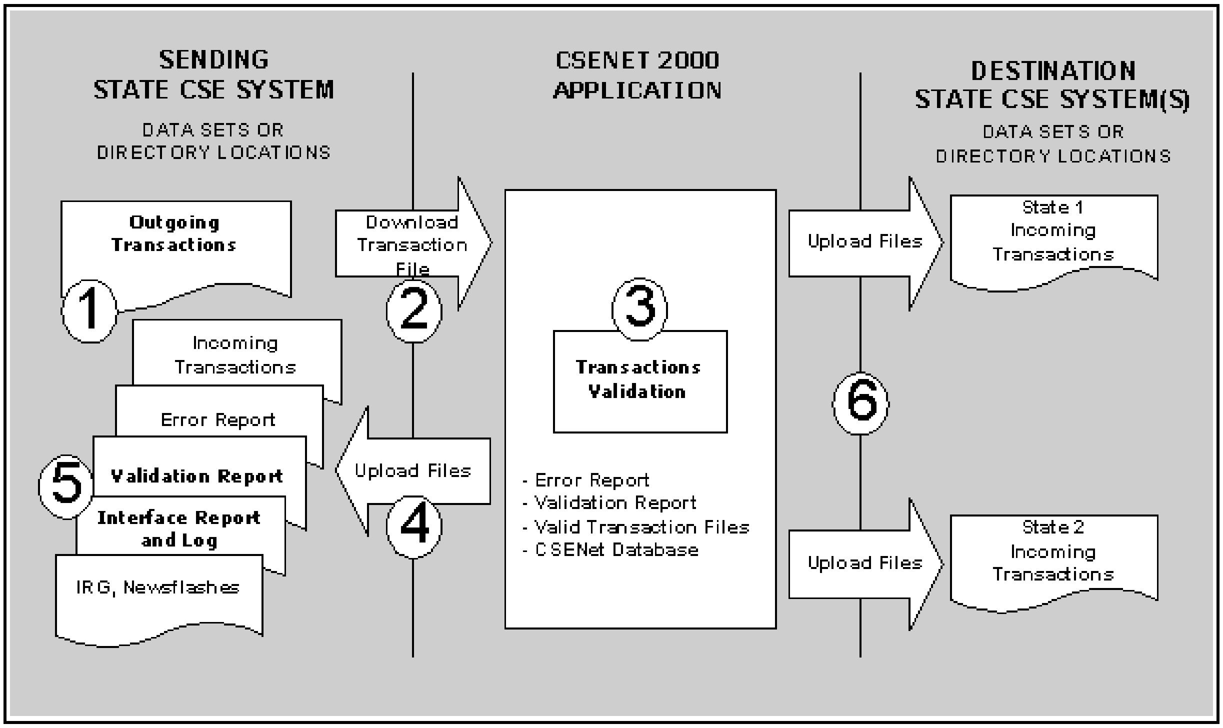
The primary data sets include:

- the Outgoing Transactions data set, containing transactions to be forwarded to other state systems;
- the Incoming Transactions data set, containing transactions retrieved from other state systems; and
- other data sets, containing Interstate Roster and Referral Guide (IRG) data or reports generated by CSENet.

(The files transferred to and from a state system by CSENet are detailed in Section 3.2.1: State Interface Application.)

Figure 3-1 displays an overview of the TEP cycle for a transaction file.

Figure 3-1: The CSENet Transaction Exchange Process



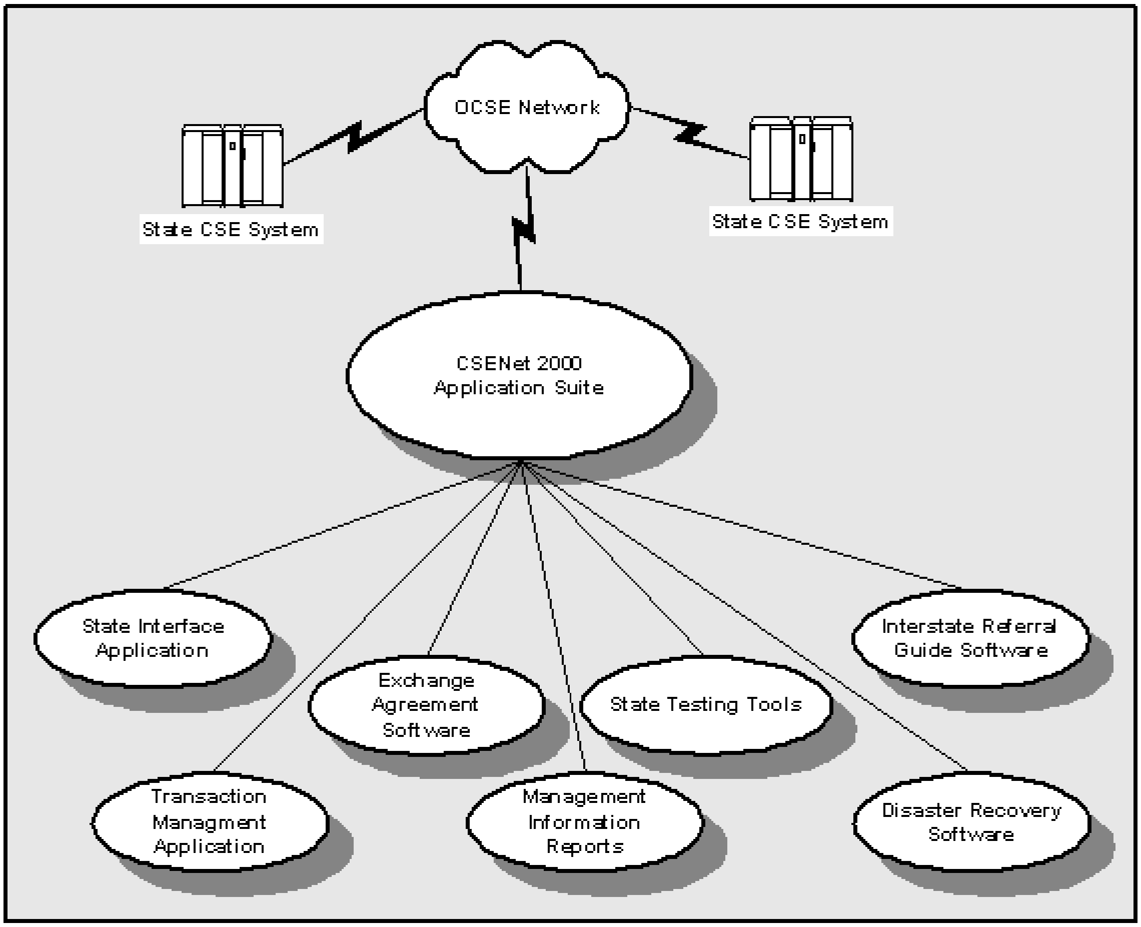
The following is a detailed step-by-step description of the TEP cycle for a transaction file.

Step 1	The sending state loads its Outgoing Transaction data set with transactions addressed to other members of the CSENet user community.
Step 2	During a pre-scheduled Send-to-State interface initiated by the OCSE server, the state's outgoing transactions are retrieved.
Step 3	Immediately after the OCSE server receives transactions, the entire transaction file is validated against the Data Block Record Layout criteria. If a validation condition is violated, an error code and message, along with the transaction's serial number, is written to the Error Report for each defective transaction. After verifying all transactions received from the state, a Validation Report is written to the Send-to-State report directory; if an Error Report exists, it is also written to this directory. Valid transactions are written to a file to be uploaded to the destination state(s) and are saved by the CSENet database. ***Note: If a transaction file did not complete processing, an error message is generated at the end of the Validation Report.***
Step 4	During a pre-scheduled Send-to-State interface session, the Validation and Error Reports are uploaded to the state's data sets defined for each of these reports. The Interface Report and log files are also uploaded during this step. Additionally, any IRG or Newsflash files the state requested are uploaded.
Step 5	States access their data set locations to obtain available reports and other data files.
Step 6	Transactions that passed validation along with any other transactions awaiting delivery are uploaded to the destination state system(s) during the pre-scheduled Send-to-State interface to the destination state(s).

3.2 Application Suite Capabilities

The application suite provides seven major capabilities to support interstate child support information processing. These components are composed of custom-developed software and off-the-shelf applications. Figure 3-2 illustrates the components.

Figure 3-2: The CSENet 2000 Application Suite


OCSE Network
State CSE System
State CSE System
CSENet 2000 Application Suite
State Interface Application
Interstate Referral Guide Software
Exchange Agreement Software
State Testing Tools
Transaction Managment Application
Management Information Reports
Disaster Recovery Software

- **State Interface Application –** provides CSENet the ability to interface with state systems;

- **Exchange Agreement Software –** provides states the ability to designate specific Function codes and states with which they desire to communicate or exchange transactions;

- **State Testing Tools –** provide states the ability to test and evaluate their CSE system programming;

- **Interstate Roster and Referral Guide (IRG) Software –** provides states semi-monthly FIPS codes updates from the IRG Web site;

- **Transaction Management Application –** provides validation and verification of state transactions and feedback to states in the form of Validation and Error Reports;

- **Management Information (M/I) Reports –** provides data primarily for ad hoc reporting.

- **Disaster Recovery Software –** provides redundancy to the operational production system during critical system outages;

These components are maintained and enhanced by the CSENet team with the support of state Technical Work Groups and OCSE. With the exception of Management Information (M/I) Reports, all capabilities are discussed in detail in this section. M/I Reports are described in Section 7.0.

3.2.1 STATE INTERFACE APPLICATION

The State Interface Application, hereafter referred to as the interface, automates the exchange of interstate child support transactions between state systems over the OCSE Network.

All files transferred via the interface require a designated location on the state system in which to be placed. On mainframes, the locations of CSENet data are data sets. On other types of servers, CSENet files are located in directories. Because CSENet files are located on a mainframe in most states, the location of CSENet files on state systems will be referred to as data sets in this document. In a few state systems, the interface data sets are in a Generation Dataset Group (GDG).

The following items must be in place on the state system to successfully execute an interface session:

- a network connection from the OCSE server to the state system, referred to as the state’s interface server;

- a userid and password to log on to the state’s interface server; and

- properly allocated interface data sets with read and write privileges granted to the userid used to log on to the state’s interface server (referred to as the state’s interface userid).

Chart 3-1 provides specifications for the interface data sets with the following guidelines:

- The Retrieve/Overwrite/Append column refers to whether the data in the data set is downloaded by the OCSE server from the state system or whether it is uploaded by CSENet to the state.
- The File Transfer Protocol (FTP) command to append data to a data set is **append**. The FTP command to overwrite a data set with new data is **put**. If a state's version of FTP does not allow the **append** command or the state does not want its interface data sets appended to, the **put** command is used for all interface uploads.
- Minimum and maximum record lengths are provided in bytes. In systems where the interface connects to a UNIX server, the record lengths are one byte longer to include a carriage return at the end of each record.

If fixed-block format is chosen for a data set, the logical record length (LRECL) defined when the data set parameters are established specifies the length of all records to be written to the data set. However, if variable-block format is chosen for a data set, its LRECL specifies the maximum length of any record written to the data set. If a record destined for a data set is longer than the LRECL, the record is either written as multiple records or truncated, depending on the FTP parameters defined on the state's interface server. The LRECL defined for each interface data set must be set to the correct length to avoid record-length errors.

CHART 3-1: INTERFACE DATA SET SPECIFICATIONS

Interface Data Set	Retrieve/ Overwrite/ Append	Minimum Record Length	Maximum Record Length	Data in the Data Set	Data Format
Outgoing Transactions	Retrieve, Then Empty	127	8481	Transactions to other states (or test transactions to send to the OCSE server).	See Appendix C: *Data Block Record Layout*.
Incoming Transactions	Append	121	8475	Transactions from other states (or test transactions from the OCSE server).	See Appendix C: *Data Block Record Layout*.
Invalid Transactions	Append	145	145	Error messages for transactions in which the OCSE server found errors.	See Appendix C: *Data Block Record Layout*.

CHART 3-1: INTERFACE DATA SET SPECIFICATIONS

Interface Data Set	Retrieve/ Overwrite/ Append	Minimum Record Length	Maximum Record Length	Data in the Data Set	Data Format
Validation Reports	Append	0	80	Validation Reports providing ***validation results including any processing errors***.	See Appendix G: *Transaction Reports*.
IRG Master File	Overwrite	247	247	A file of all existing IRG records.	See Appendix F: *IRG Data Format.*
IRG Update Files	Append	247	247	A file of new or modified IRG records.	See Appendix F: *IRG Data Format*.
Newsflashes	Append	0	80	Text files containing information for CSENet users.	Alphanumeric text.
Interface Reports	Append	0	80	Interface Reports providing summary information about interface file transfers.	See Appendix H: *State Interface Reports.*
Interface Logs	Append	0	120	Interface Logs containing detailed output from interface file transfers (useful for debugging).	Alphanumeric text.

3.2.1.1 Retrieve-from-State Interface

The interface operates in two directions: Retrieve-from-State and Send-to-State. In a Retrieve-from-State interface, CSENet data is downloaded from a state system.

3.2.1.1.1 DATA TRANSFER

The file transfers in the Retrieve-from-State interface are executed in the following order:

1. Download transactions to be sent to other states from the Outgoing Transactions data set using the FTP download command **get**.

2. If no file transfer error occurred, upload an empty file into the Outgoing Transactions data set. (See Section 3.2.1.1.2 *Transfer Errors* on page 3-8 for information on errors.)

Step 2 is performed to prevent receiving duplicate transactions from the state system during the next Retrieve-from-State interface session. Besides reading the Interface Report or Interface Log, the only way for a state to know that the outgoing transactions were successfully downloaded by the interface is when the Outgoing Transactions data set contains zero records after the interface session is completed. (The specifications for the Outgoing Transactions data set are provided in Chart 3-1: *Interface Data Set Specifications* on page 3-6.)

3.2.1.1.2 TRANSFER ERRORS

During each Retrieve-from-State interface session, the interface downloads data from the Outgoing Transactions data set. If an error occurs while downloading and the state system appends to the data set rather than overwriting it each day, no action is necessary for the OCSE server to receive all the state’s outgoing transactions during the next Retrieve-from-State interface session. On the other hand, if the state system overwrites the data set each day, any data not forwarded must be combined into a single file for the interface to download. The interface is not designed to download multiple cycles of GDG data sets.

After the interface successfully downloads data from the Outgoing Transactions data set, it replaces the data with an empty file. If an error occurs while emptying the data set, any transactions downloaded from the state system are archived, but are not processed. This prevents processing of duplicate transactions. Note: The Outgoing Transactions data set should never be emptied by the state system.

Chart 3-2: Interface Error Sources and General Solutions on page 3-11 contains a list of interface errors and their resolutions.

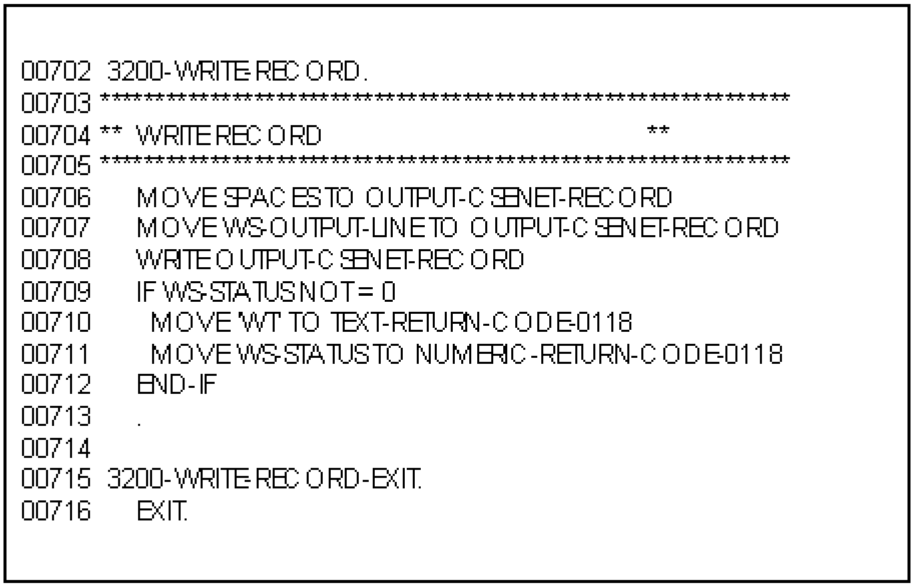
3.2.1.1.3 DELIMITING OUTGOING TRANSACTION RECORDS

Each record written to the Outgoing Transactions data set must end with a new line character (octal `'\012'` or character `'\n')`. Without a new-line character, any spaces at the end of a record will be stripped off by the OCSE server. A record containing less than the number of bytes specified for its type in the Data Block Record Layout will be found invalid during the validation process.

It is recommended that the new-line character in each record in the Outgoing Transactions data set be placed in the byte directly after the last data block in the record. A less-preferable alternative is to pad each record with spaces until the LRECL for the data set is reached and then enter the new-line character. However, because the spaces are transferred as part of each record, this method consumes more space on the OCSE server. For example, a transaction with only a Header should be 127 bytes; but, if padded with spaces up to the LRECL, the downloaded record consumes 8481 bytes.

Figure 3-3 shows code excerpted from a state's COBOL program that writes records to the Outgoing Transactions data set with a new line character.

Figure 3-3: Sample Code to End Records with a New Line Character



```
 3200-WRITE-RECORD.
************************************************************
** WRITE RECORD                                          **
************************************************************
    MOVE SPACES TO OUTPUT-CSENET-RECORD
    MOVE WS-OUTPUT-LINE TO OUTPUT-CSENET-RECORD
    WRITE OUTPUT-CSENET-RECORD
    IF WS-STATUS NOT = 0
      MOVE 'WT' TO TEXT-RETURN-CODE-0118
      MOVE WS-STATUS TO NUMERIC-RETURN-CODE-0118
    END-IF
    .

 3200-WRITE-RECORD-EXIT.
    EXIT.
```

3.2.1.2 Send-to-State Interface

In a Send-to-State interface session, CSENet data is uploaded to rather than downloaded from a state system.

3.2.1.2.1 DATA TRANSFER

The file transfer steps in the Send-to-State interface are executed in the following order:

1. Upload any unsent transactions from other states to the Incoming Transactions data set.
2. Upload any unsent invalid transaction error messages to the Invalid Transactions data set.
3. Upload any unsent Validation Reports to the Validation Reports data set.
4. Upload any unsent IRG Master data to the IRG Master File data set.
5. Upload any unsent IRG Update data to the IRG Update File data set.

6. Upload any unsent Newsflashes to the Newsflashes data set.
7. Upload any unsent Interface Reports to the Interface Reports data set.
8. Upload any unsent Interface Logs to the Interface Logs data set.

Parameters for the Incoming Transactions data set must be established (allocated) by the state in order to receive transactions from another state. If the state sends transactions to the OCSE server, the Invalid Transactions and Validation Reports data sets should be allocated by the state in order to receive any transaction error messages detected by CSENet during processing. The other data sets provide additional information, but are not essential for minimal communications. Note: If any of these data sets are not allocated by the state or the privilege to create them is not granted to the state's interface userid, no file is uploaded into the data set. Data set specifications are shown in Chart 3-1: *Interface Data Set Specifications* on page 3-6.

3.2.1.2.2 TRANSFER ERRORS

If an upload error occurs during a Send-to-State interface session, the entire file containing the untransferred data remains in the outgoing data directory on the OCSE server so that it can be uploaded in the next Send-to-State interface session. (See Chart 3-2 on page 3-11 for a list of interface errors and their resolutions.)

3.2.1.2.3 CREATE VS. APPEND-TO-STATE DATA SETS

In most cases, the interface appends new data to the end of each of the state's Send-to-State interface data sets with the exception of the IRG Master File data set. (The IRG Master File data set is overwritten in all states so that it never contains duplicate records.) Appending to a data set ensures that the interface does not overwrite any data before the state system is able to process it.

The FTP command to append data to a data set is **append**. The FTP command to overwrite a data set with new data is **put**. If a state interface server's version of FTP does not allow the **append** command or the state does not want its data sets appended to, the **put** command is used for all uploads.

3.2.1.3 Interface Troubleshooting

Interface results are written to three files for each execution of the interface session. These three files, the Interface Log, Interface Report, and Interface Summary are described in detail below:

- An Interface Log is created with detailed output about the ping(s), logons, and file transfers during the interface session.

- An Interface Report is created with summary information about interface file transfers and any errors encountered during the interface session.

- A one-line Interface Summary, plus any interface errors, is written to a Server Interface Report used by the CSENet team for system management.

3.2.1.3.1 DETECTING AN INTERFACE PROBLEM

All interface errors are reported in a Server Interface Report. This report provides the initial indication of an interface error.

A number of errors are usually encountered when setting up the interface to a state system. During the first attempts at establishing interface sessions with a state system, the CSENet team reviews the Interface Logs generated by the sessions and works closely with states to resolve errors. Until the initial errors are resolved, CSENet does not consider the interface complete. Refer to Section 8 for details on support available to states.

3.2.1.3.2 DIAGNOSING AND SOLVING AN INTERFACE PROBLEM

For interface errors specific to a Retrieve-from-State interface or a Send-to-State interface, consult Section 3.2.1.1.2: Transfer Errors on page 3-8 and 3.2.1.2.2: Transfer Errors on page 3-10. Chart 3-2 provides the general sources of and solutions for interface errors.

CHART 3-2: INTERFACE ERROR SOURCES AND GENERAL SOLUTIONS

Interface Error Source	General Solution to the Error	Who Implements
Network or connection-related errors	Analyze the network connection if the problem continues. Verify that a connection to the state server is possible from the OCSE router.	State/ CSENet
Incorrect State Profile information	Provide correct State Profile information to the state's CSENet technical representative.	State
Log on or data set specification errors on the state system.	Follow the specifications and recommendations outlined in this document.	State

The following comments should be noted when analyzing interface errors and determining their solution:

1. The only differences among the interface communication with different state systems are the State Profile parameters for the state's mainframe IP address, logon userid and password, and data set names. Except for the definition of these parameters, the interface functions the same with respect to all state systems.

2. The CSENet team is available to aid state users and engineers in solving interface problems. However, most interface problems require actions by state personnel for resolution. A teleconference involving all related parties is often required to solve a long-standing problem.

3. If an interface error is received, the CSENet technical representative or the Service Desk can be contacted to request a manual re-execution of the interface session. Some errors, such as logon errors, require resolution before a manual session can be initiated.

4. Due to varying volumes of network traffic on state servers, it is normal for connection-related errors to occasionally occur in interface sessions. However, if connection problems persist, contact the CSENet Service Desk. (See Section 8 for information on support available to states.)

5. If ping, connection, or certain data set errors persist in interface sessions to a state, it may be appropriate for the state to request a change to the interface execution time to reduce conflicts with user needs or other processes running on the state system.

6. To prevent password errors, a non-expiring password or a password containing part of a numeric date is recommended. Refer to Figure 3-2 for an example of a password containing a date.

7. Many data set errors are caused by improperly-written or lack of scripts that create and process the data sets. The most common errors are caused by:

   - incorrect LRECL parameters;
   - failure to grant Resource Access Control Facility (RACF) privileges to the state's interface userid to read and write to the data sets; and,
   - data set name differences between the state system and the State Profile.

Each interface data set must be correctly configured (allocated) prior to an interface session or the state system's default parameters will be used. The correct LRECL parameters for CSENet transactions are 8481 for an outgoing transaction and 8475 for an incoming transaction. In most states, the default LRECL for unallocated data sets is either 80 or 128 bytes.

For example, if the interface writes data to an Incoming Transactions data set that has not been allocated by the state, all records will be written to the data set with the system's default LRECL. Each record will either be written as multiple records (wrapped) or truncated, depending on the state system.

1. The state must define all data set parameters such as LRECL and variable- or fixed-block format. Note: The OCSE server does not configure state data set parameters.

2. Unless there is a ping, connection, or logon error or an error with the Interface Report and/or Interface Log data sets, each state receives a State Interface Report and State Interface Log with each Send-to-State interface session. Any error messages received on Interface Reports are error messages from the State Interface Log, reworded for greater clarity. See Appendix H for an example of a State Interface Report.

3. If more than one transaction file was received from the state since the last successful Send-to-State interface session, there will be two reports in the state's Validation Reports and/or Interface Reports data sets.

Appendix O: *State Interface Errors and Resolutions* lists errors as they are reported on the State Interface Report and the Server Interface Report, grouped by type of error. Although some of the errors cited are rare, all known interface errors are included for completeness.

3.2.1.4 State Profile Interface Parameters

The State Interface Profile on the OCSE server contains all information required for the interface to a state. The State Interface Profile is obtained directly from the State Profile provided by the state point-of-contact or technical point-of-contact. This section provides direction on how to complete the interface-related portions of a State Profile. An example is included in Appendix I: *State Profile*.

For the OCSE server to successfully transfer data to and/or from a state's interface data set, the following actions must have been completed on the state's interface server:

1. On mainframes, the data set must be properly allocated, usually with a Job Control Language (JCL) script; and

2. In all states, the necessary read-and-write privileges must be granted on the data set to the userid used to log on to the state's interface server.

If a data set has not been allocated, but the state's interface userid has the permission to create it on the state's server, the interface will write data to the data set, thereby creating it.

However, the default state system parameters are used, since the data set was not allocated. (See comment 7 on the previous page for additional information and Chart 3-1: Interface Data Set Specifications on page 3-6 for maximum record lengths).

3.2.1.4.1 CSE SYSTEM LOGON PARAMETERS

The CSENet suite uses File Transfer Protocol (FTP) for transferring files between state CSE systems. In order for the OCSE server to exchange data with the state's CSE system, it must be granted access. States grant access to the OCSE server by providing logon information (i.e., userid and password).

The following are directions for completing the Logon Using FTP section of the State Profile:

- For User-ID, specify the userid the interface is to use to log on to the system.

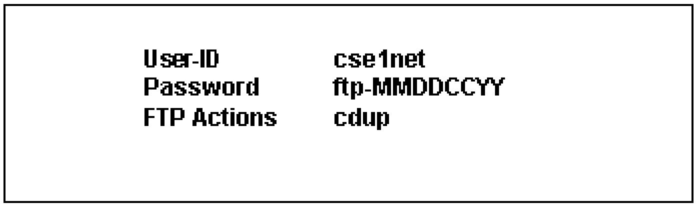
- For Password, specify the password the interface is to use to log on to the system.

- FTP Actions should only be completed in states where a change directory (**cd**) command is required after logging on to the state system. When an FTP transfer to an IBM mainframe is performed, data set names are usually prefixed with the logon userid, unless a **cdup** command is executed before any file transfers. A **cd** command, e.g. **cd test**, is often needed when the interface server is on a UNIX platform. The interface only allows one **cd** command for a state's production data sets and one for test data sets.

A non-expiring interface password or a password containing a date is recommended to prevent password errors. For example, a JCL script can be created to change the interface password on the first of each month, e.g. in January the password might be **cse_01** and in March it would be **cse_03**. The interface is able to figure the date portion of a password when it executes a session. When writing a script to change an interface password using part of the date, it is necessary to account for the time zone difference between CSENet (Eastern) and the state.

Figure 3-4 illustrates an example of the parameters in the Logon Using FTP section of the State Profile. In the password, MMDDCCYY is the month, day, century, and year.

Figure 3-4: Sample Logon Using FTP

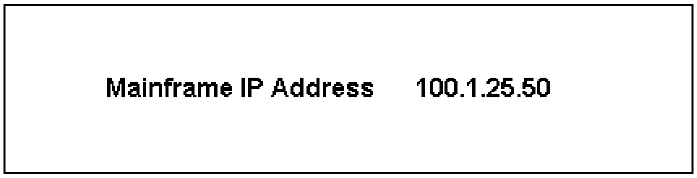


User-ID	cse1net
Password	ftp-MMDDCCYY
FTP Actions	cdup

3.2.1.4.2 CSE SYSTEM IP ADDRESSES

The IP address for the state's interface server is listed as the Mainframe IP Address in the State Profile. All IP addresses on the State Profile must be provided in dotted decimal (not Domain Name System) format, since dotted decimal is the current standard for the OCSE server. Figure 3-5 shows an example of a server's IP address in the IP Addresses section of the State Profile.

Figure 3-5: Sample Interface Server IP Address



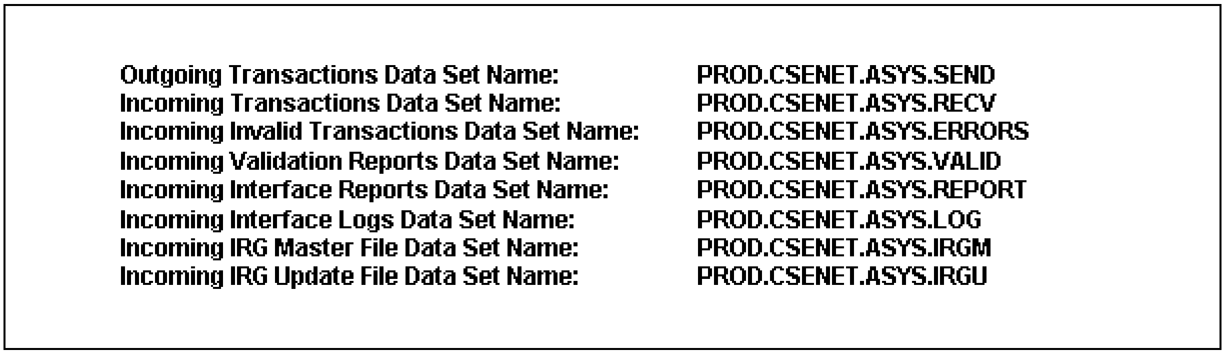
Mainframe IP Address 100.1.25.50

3.2.1.4.3 CSENET 2000 DATA SETS

Interface data sets are listed in the Production Data Set Names and Test Data Set Names sections of the State Profile. Two sets of data sets are needed so that production and test data remain separate. When specifying data set names on the State Profile, use the data set naming conventions required by the state system.

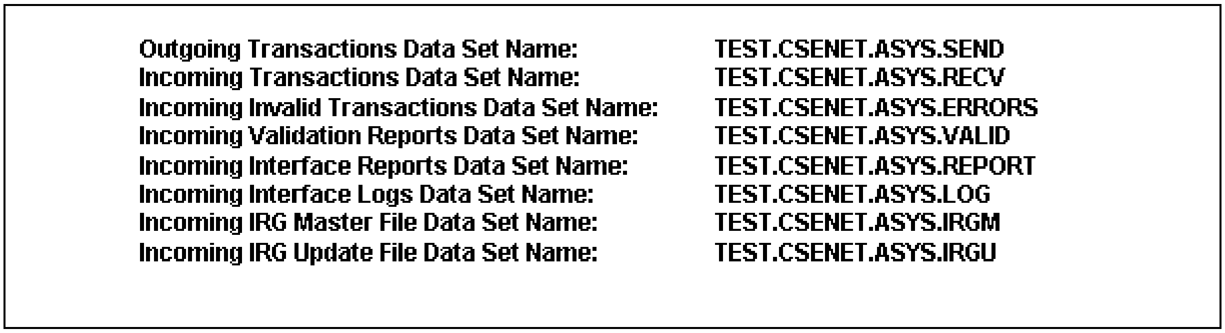
Figures 3-6 and 3-7 show examples of the parameters for Production Data Set Names or Test Data Set Names on a State Profile.

Figure 3-6: Sample Production Data Set Names



Outgoing Transactions Data Set Name:	PROD.CSENET.ASYS.SEND
Incoming Transactions Data Set Name:	PROD.CSENET.ASYS.RECV
Incoming Invalid Transactions Data Set Name:	PROD.CSENET.ASYS.ERRORS
Incoming Validation Reports Data Set Name:	PROD.CSENET.ASYS.VALID
Incoming Interface Reports Data Set Name:	PROD.CSENET.ASYS.REPORT
Incoming Interface Logs Data Set Name:	PROD.CSENET.ASYS.LOG
Incoming IRG Master File Data Set Name:	PROD.CSENET.ASYS.IRGM
Incoming IRG Update File Data Set Name:	PROD.CSENET.ASYS.IRGU

Figure 3-7: Sample Test Data Set Names



Outgoing Transactions Data Set Name:	TEST.CSENET.ASYS.SEND
Incoming Transactions Data Set Name:	TEST.CSENET.ASYS.RECV
Incoming Invalid Transactions Data Set Name:	TEST.CSENET.ASYS.ERRORS
Incoming Validation Reports Data Set Name:	TEST.CSENET.ASYS.VALID
Incoming Interface Reports Data Set Name:	TEST.CSENET.ASYS.REPORT
Incoming Interface Logs Data Set Name:	TEST.CSENET.ASYS.LOG
Incoming IRG Master File Data Set Name:	TEST.CSENET.ASYS.IRGM
Incoming IRG Update File Data Set Name:	TEST.CSENET.ASYS.IRGU

3.2.1.4.4 MODIFYING INTERFACE PARAMETERS

Modifications to an interface parameter on the state system may be performed at any time. However, the changes must be reported to the CSENet Service Desk and implemented at the same time to ensure successful execution of the file transfer process. See Section 8.0: *Technical Support for States* for instructions on reporting this information.

3.2.1.5 Executing Interface Sessions

Interface sessions may be executed with a state system in one of three ways:

- Execution during one of the automated execution times listed in Chart 3-3. This accounts for the majority of production file transfers.

- Execution at a specific time based on a request from a state. State contacts often request automated interface sessions to pick up and drop off data and to test data

- Execution of a manual transfer based on special state circumstances.

3.2.1.5.1 AUTOMATED INTERFACES

After a new or modified State Profile is received from a state, the information in the profile is verified either with manual or automatic interface sessions. The success or failure of the test(s) is discussed with state staff and any necessary changes are determined.

Once the interface connection to a state is successful, at least in the logon to the state system, the interface to the state is activated on the OCSE server in production or testing mode, depending on the state's request.

Incomplete programming should not prevent a state from establishing connection to the interface, because any data can be used to test the interface. Since interface testing only confirms the ability of the interface to log on to a state system and to read and write to the state's data sets, the interface does not test the validity of data. It is advantageous to know that once the state's CSENet programming is complete, the interface to the state is established.

The state can also receive non-transaction data files prior to going into production (i.e., IRG files, Newsflashes, bulletins, transaction test files, and Interface Reports).

Data received from interface testing should be closely reviewed to confirm that it arrived in the expected format. For example, record-length errors are very common when first establishing the interface to a state.

3.2.1.5.2 PRODUCTION INTERFACE SCHEDULE

Chart 3-3 shows the scheduled execution times for automated interface sessions to production data sets. Each state informs the CSENet technical representative of its preferred execution group. The criteria for selecting the interface execution time is usually based on avoiding periods of contention with user needs and other processes on the state CSE system.

CHART 3-3: SCHEDULED EXECUTION TIMES FOR AUTOMATED INTERFACES

Interface Group Time	Interface Description	Number of States in This Group as of June, 19, 2003
1-1:30AM Eastern	Retrieve-from-State	30
4-4:30AM Eastern	Send-to-State	30
1-1:20PM Eastern	Retrieve-from-State	24
4-4:20PM Eastern	Send-to-State	24

Unless a state requests that the interface be turned off, an automated interface session occurs every weekday of the year. There may be days when the data sets are not processed by a state, for example, holidays, but the interface sessions occur automatically nevertheless.

3.2.1.5.3 MANUAL INTERFACE REQUESTS

Upon request from a state contact, an interface session may be manually executed. If regular requests for manual interface sessions are anticipated, it is preferable to consider establishing an automated execution time, since no human intervention is required.

3.2.1.6 State Data Sets

This section provides guidelines for allocating and processing CSENet interface data sets.

3.2.1.6.1 GDG VS. NON-GDG DATA SETS

This section discusses the logic for processing interface data sets on a state system. The logic provided here considers how the interface reads and writes data from and to interface data sets (for details on file transfers, see the two sections on data transfer: Section 3.2.1.1.1 on page 3-8 and 3.2.1.2.1 on page 3-9).

Creating non-GDG interface data sets is preferable to using GDGs. When GDGs are used, human intervention is usually required when an interface session fails. Creating a non-GDG Outgoing Transactions data set enables all unretrieved data to be retrieved from the state during each Retrieve-from-State session. Otherwise, if two cycles of a GDG Outgoing Transactions data set need to be retrieved, manual steps must be taken to combine the data in them. Creating non-GDG Send-to-State data sets ensures the state system never has more than one incoming file to process, because the interface appends new data to any old data.

If the state creates a GDG for a Send-to-State interface data set, it must devise a method to process multiple cycles of the data set.

3.2.1.6.1.1 Retrieve-from-State Non-GDG Data Set

The following steps outline a process for the state system program, usually a JCL script, that creates a non-GDG Outgoing Transactions data set:

1. Create a GDG data set for new outgoing transactions.
2. Combine the new data with any unsent data in the Outgoing Transactions data set, usually by appending to the data set.

Using these steps, the CSENet interface is able to pick up multiple days of transactions from the Outgoing Transactions data set and the state retains the benefits of GDG data sets for archiving.

If the download in a Retrieve-from-State interface session fails, any unretrieved data will be retrieved in the next session. The interface uploads an empty file into the Outgoing Transactions data set after successfully downloading from it, and thereby removes any need for the state to generate procedures to handle duplicate data.

3.2.1.6.1.2 Send-to-State Non-GDG Data Sets

The following steps outline a standard process for each state system program, usually a JCL script, that processes and recreates a Send-to-State interface non-GDG data set each day:

1. If there is data in the data set, copy it to a GDG data set.
2. Reallocate the interface data set using the specifications in Chart 3-1: Interface Data Set Specifications on page 3-6, or delete all records from it.
3. Process the GDG data set.

Using these steps, the state system will never have more than one file to process, but it can still archive each file as a GDG.

Whether the state allocates GDG or non-GDG data sets, the data sets for the Interface Reports and Validation Reports can contain multiple reports if more than one transaction file was received from the state since the last successful Send-to-State session to the state. In order to process these data sets, states must devise a method for distinguishing between multiple reports.

3.2.1.6.2 ALLOCATING DISK SPACE FOR EACH DATA SET

Each data set requires storage space on the state's interface server. The amount of space needed varies from state to state. The key factors affecting space requirements are:

- the volume of transactions the state expects to process;
- the state's average expected transaction size; and
- how often the state processes the interface data sets.

Chart 3-4 provides examples for calculating data set space requirements. For Reports, Logs, and Newsflashes, the number of records per day equates to the number of lines in the file written to the data set. Sufficient space should be allocated to each data set to store multiple days of data.

CHART 3-4: SAMPLE INTERFACE DATA SET SPACE ALLOCATIONS

Interface Data Set	Maximum Record Length	Number of Records Per Day	Number of Days to Be Able to Hold Data	Number of Bytes of Storage Space to Allocate
Outgoing Transactions	8481	2000	5	84810000
Incoming Transactions	8475	2000	5	84750000
Invalid Transactions	145	40	31	179800
Validation Reports	80	60	31	148800
IRG Master Files	148	10000	1	1480000
IRG Update Files	148	3000	4	444000
Newsflashes	80	1000	4	320000
Interface Reports	80	50	31	124000
Interface Logs	120	200	31	744000

3.2.1.6.3 TESTING VS. PRODUCTION DATA SETS

Two sets of interface data sets are needed in each state so that production and test data are never intermingled. These data sets are listed on the State Profile as Production Data Set Names and Test Data Set Names.

When the interface executes a session, a command-line switch is used to determine whether the interface is to transfer files to and from the state's production data sets or its test data sets. Production data is stored on one OCSE server and test data on another server.

3.2.1.7 Data Archiving

This section discusses the archiving of CSENet data on the state system and the OCSE server.

3.2.1.7.1 STATE CSE SYSTEM

Most states choose to archive CSENet data for a specified number of days, or cycles of GDGs, in accordance with state and federal requirements for archiving data. The amount of interface data and the number of days to archive needs to be determined by state system staff. The list in Chart 3-5 provides general guidelines for the minimum number of days to archive CSENet data on a state system.

CHART 3-5: INTERFACE DATA STORAGE ON THE STATE SERVER

Data	Minimum Archive Period
Transaction data, invalid transaction data, and Validation Reports	5–31 days, or GDG cycles
Interface Reports and Interface Logs	10–31 days, or GDG cycles
IRG Master file	1 day, or GDG cycle
IRG Update files	4 days, or GDG cycles
Newsflashes	1 day, or GDG cycle

3.2.1.7.2 OCSE SERVER

All CSENet data uploaded and downloaded to a state system is archived by the CSENet suite for at least 30 days. Transaction data is stored in the database for at least 90 days. Backups of the operating system and database are performed daily. In addition, CSENet data is ported to the Backup server daily for use in the event of a critical system failure on the Production server.

Chart 3-6 shows the minimum number of days each type of interface file is stored on the OCSE server.

CHART 3-6: INTERFACE DATA STORAGE ON THE OCSE SERVER

Interface File Type	Minimum Number of Days Files are Archived
Outgoing Transactions	120
Incoming Transactions	120
Invalid Transactions	90
Validation Reports	90
IRG Master Files	90
IRG Update Files	90
Newsflashes	90
Interface Reports	90
Interface Logs	120

3.2.2 TRANSACTION MANAGEMENT APPLICATION

The Transaction Management Application (TMA) is a group of functions in the CSENet suite used for the validation of the Retrieve-From-State transaction file, database loading, and routing of CSENet transactions.

3.2.2.1 Transaction Validation Process

The transaction validation process enforces data integrity and promotes commonality throughout the CSENet user community. The TMA verifies each transaction received from a state system against the Data Block Record Layout rules, constraints, and data format standards (See Appendix C).

If the transaction passes validation, it is loaded into the CSENet database and routed to a file to be uploaded to the destination state system. If the transaction encounters errors at any point during the validation process, it is rejected and an error message is added to the Error Report file (described in Section 3.2.2.1.3 on page 3-23). After all transactions in a file are processed, the TMA generates a Validation Report. In the next Send-to-State interface session, the Validation Report file and Error Report file are uploaded to the data sets designated for these reports.

Note: If a transaction file did not complete processing within the allotted time frame or was partially processed, the TMA generates a processing error message at the end of the Validation Report.

3.2.2.1.1 TRANSACTION FILE

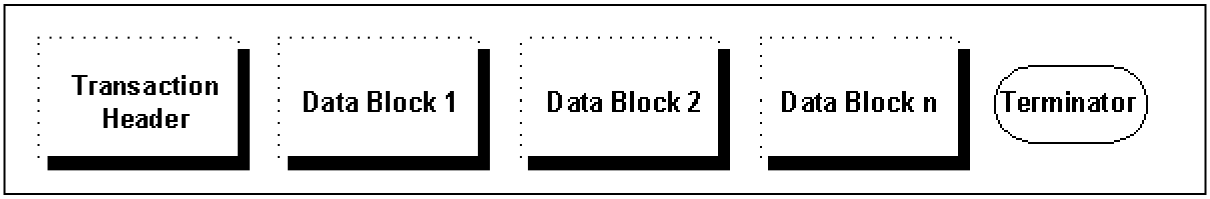
A transaction file is a flat ASCII file containing transactions received from a state system. Each transaction occupies one record in the file and each record is terminated with a new line character (octal `'\012'` or character `'\n'`).

Figure 3-8 depicts the format of a transaction record. Every transaction record must begin with a Header and end with a new line character. The Header provides key transaction information such as:

- the source and destination (FIPS) code;
- transaction serial number to uniquely identify the transaction;
- transaction type and action code, the case identification number(s) to which the transaction refers; and
- data block indicators that indicate how many data blocks, if any, are attached to the Header.

For a more detailed description of the layout of a transaction Header, refer to Section 5.0 *Transaction Structure*.

Figure 3-8: CSENET 2000 Transaction Layout



The transaction data blocks contain case data transmitted from one state to another. One or more data blocks may follow the transaction Header. The minimum number of data blocks that must be transmitted depends on the type of transaction being sent. However, a state may attach additional data blocks to any transaction. There are several kinds of data blocks that can be attached to a transaction.

Chart 3-7 describes the data blocks and indicates the maximum number of data blocks that may be attached to a single transaction.

CHART 3-7: CSENET 2000 DATA BLOCKS

Data Blocks	Maximum Number	Byte Size	Description
Case	1	351	General case information, contact, and payment addresses
NCP Identification	1	181	Physical description of the noncustodial party
NCP Locate	1	1421	Location and employer information about the noncustodial party
Participant	9	341	Information about other participants in the case

CHART 3-7: CSENET 2000 DATA BLOCKS

Data Blocks	Maximum Number	Byte Size	Description
Order	9	254	Support or paternity order information
Collection	9	70	Information about order collections
Information	1	416	General text information

Note that, while not every data block appears in every transaction (in fact, most transactions will have only a few data blocks), the data blocks must appear in the order shown in the table above. For example, when a transaction containing a Header, a Case data block, and an Information data block is generated, the Information data block must come after the Case data block, not before it.

3.2.2.1.2 VALIDATION REPORT

The Validation Report shows error statistics for a state's transaction file that has been through the validation process described in Section 3.2.2.1 on page 3-21. A sample Validation Report can be viewed in example G-1 of Appendix G. The report contains the following data elements:

- the date and time the report was created;
- the name (Local-FIPS-State) and byte size of the file validated for errors;
- the number of transactions, valid transactions, invalid transactions, and number of errors in the file validated;
- error message statistics grouped by the type of errors received; and
- error message statistics grouped by the data block in which the errors occurred.

In the event of a TMA processing error, the Validation Report also contains an error message to inform the state that the transaction file is still being processed. A sample of this message can be viewed in Appendix G.

3.2.2.1.3 TRANSACTION ERROR REPORT

The Transaction Error Report, referred to as the Error Report, contains error messages for each transaction error detected during the transaction file validation process. Error messages written to the Error Report are based on the record layout specified in Appendix M and use the error codes and messages listed in Appendix E. A sample report is presented in example G-3 of Appendix G. The Error Report is uploaded to the Invalid Transactions data set of the state that sent the transaction(s).

Multiple error messages may be generated for a transaction unless the transaction contains a fatal error. After the first fatal error is identified, the Validation process for that transaction stops and the fatal error is written to the Error Report. Chart 3-8 displays all potential fatal errors in a CSENet transaction.

CHART 3-8: POTENTIAL FATAL ERRORS IN A CSENET TRANSACTION

Error Code	Message	Description
E203	Duplicate Transaction- Check Trans Serial Num	The Transaction-Serial-Number is a duplicate, i.e. the state has already used it in a transaction previously received from the state.
E204	Communications not allowed	Communications are not allowed, i.e. an Exchange Agreement has not been enabled with the state defined in the Other FIPS State field for the Function code received.
E301	Invalid Functional type code	Functional-Type-Code must equal MSC, CSI, ENF, EST, COL, PAT, or LO1.
E831	Invalid Combination of FunctTypeCode, Act Code, and Act Reason	The combination of Functional-Type, Action, and Action-Reason Code is invalid.
E904	Invalid Other FIPS State	The state defined in the Other-FIPS-State field is not a valid FIPS state according to the IRG.
E905	Invalid Other FIPS County	The County defined in the Other-FIPS-County field is not a valid FIPS County according to the IRG.
E906	Originating FIPS does not match Local FIPS code	The state FIPS from which the transaction was received does not match the value in the Local-FIPS-State field.
E907	Invalid Local FIPS State Code	The state defined in the Local-FIPS-State field is not a valid FIPS state according to the IRG.
E908	Invalid Local FIPS County Code	The County defined in the Local-FIPS-County field is not a valid FIPS County according to the IRG.
E935	Case ID Reconciliation Comm. Not Allowed	No valid Case-ID exchange agreement exists.

3.2.2.2 Transaction Loading and Routing

The transaction loading and routing functions of CSENet load valid transactions into the OCSE server's database and route them to the appropriate state systems. Transactions are routed to the state specified in the Other-FIPS-State field in the transaction's Header.

3.2.3 STATE TESTING TOOLS

Several capabilities in the CSENet suite are dedicated to testing the programming in a state system. These tools include the Interface-to-Test data sets on a state system, the Test Deck Application, state loopback testing, and the transaction analysis tools.

The Interface-to-Test data set is used to pick up test transactions from the Outgoing Transactions test data set in a state system and to return the Validation and Error Reports generated by processing these transactions. Any valid transactions from the state with a destination FIPS of another state are routed to a file to be sent to the Incoming Transactions test data set in the destination state system. If there is an automated interface set up for the destination state, the transactions are sent during the next automated session. However, if an automated interface is not established, either the user in the state that generated the transactions or a user in the destination state must request a manual interface session from the CSENet Service Desk.

Any valid test transactions with a destination FIPS code of 9100000 or 9500000 are archived by CSENet, but are not sent to any state. These FIPS codes are used exclusively for a state to test transaction validity with CSENet, not for testing with other states.

The Test Deck Application tests the ability of a state system to process transactions it receives. Upon request, the Test Deck can be used to generate a file of transactions to be sent to the Incoming Transactions test data set. The Test Deck can generate files with the following types of transactions:

- one of each type of all valid transactions with varying Function, Action, and Reason code combinations;

- one or more maximum length transactions containing the maximum number of all types of data blocks; or

- any number of valid transactions based on a valid transaction Function, Action, and Reason code combination.

Appendix N contains a complete list of predefined test scenarios that are available on request, along with the expected results of the tests.

In addition to the above tools, another test tool available to states is loopback testing. Loopback testing provides the ability to generate tailored transactions to suit a state's specific needs. For example, states create transactions using their own FIPS code as both the Local-FIPS-State and Other-FIPS-State codes. Once the OCSE server receives the transactions, it performs validation and returns the appropriate information to the originating state for analysis. Whenever desired by states, loopback testing may be performed; however, it can only be used with test data sets.

3.2.3.1 Required Test Data Sets

Two sets of interface data sets are needed in each state, one for production and one for testing. These data sets are listed on the State Profile as Production Data Set Names and Test Data Set Names.

Production data is stored on one OCSE server and test data is stored on another. The advantages of having test data sets include:

- preserving data integrity by separating test from production data;
- ease of testing between CSENet and other states; and
- easier analysis of test files.

Figure 3-7 on page 3-16 illustrates an example of test data set names. These are based on the specifications for production data sets shown in Figure 3-6 on page 3-15.

3.2.3.2 Transaction Analysis

The Transaction Analysis tools were developed to aid states and the CSENet team in analysis of transactions. The uses of this analysis software include the following:

- tracking patterns about the usage of CSENet transactions to aid in developing future CSENet requirements and enhancements;
- determining the cause of invalid transactions to aid state users in modifying their state CSENet programming; and
- provide support to states for transaction error resolution.

When requesting transaction analysis, please include the following information:

- the Local-FIPS-State code;
- the Other-State-FIPS code in the transaction;
- the date the transaction was picked up by CSENet;
- the transaction serial number; and
- the field and data block in question.

See Section 8.0: *Technical Support for States* for information on requesting technical support.

3.2.3.3 Test File Transmission

To request test file transmission to or from a state system, states should contact the CSENet technical representative and specify the desired test scenario(s) that appear in Appendix N. The technical representative coordinates testing with CSENet staff and other states as needed or requested. Note: Test transactions should not be submitted to the production system. See Section 8.0 for additional end-user support available to states for testing.

3.2.4 EXCHANGE AGREEMENT SOFTWARE

The Exchange Agreement software provides states the ability to exchange production and test transactions. State representatives can contact their CSENet technical representative to establish an agreement to exchange specific Function code(s) with another state. When there is a need to stop communications, states have the following options;

- Disable with specific states;
- Disable by Function code; or
- Disable by specific state and Function code.

3.2.4.1 Enabling State Exchange Agreements

To establish an Exchange Agreement, the point-of-contact from the originating state should contact the point-of-contact in the other state directly. Once the Exchange Agreement for a specific Function code(s) is verified, communications are enabled to allow the exchange of transaction information between the two states. Communications can be established for either the production or test systems.

3.2.4.2 Disabling State Exchange Agreements

To disable communications with one or more states, the state’s point-of-contact should contact the CSENet technical representative to specify the Function code(s) and state(s) with which to discontinue communications. Additionally, the point-of-contact should inform the point(s)-of-contact in the state(s) with which communications are being disabled to prevent the generation of further transactions. More important, if a state’s system is going to be out of service for an extended period, the point-of-contact should request that all Exchange Agreements be disabled.

3.2.4.3 Communications Status

States can view the communications status for their state and other states on the IRG Web site.

3.2.5 INTERSTATE ROSTER AND REFERRAL GUIDE SOFTWARE

The Interstate Roster and Referral Guide (IRG) is an information resource tool used to facilitate the exchange of information relevant to child support enforcement between states.

IRG data includes states' profiles of services and valid FIPS codes and addresses, federal and regional office data, and demographic data on international child support enforcement agencies.

The IRG information relevant to CSENet consists solely of state FIPS codes and address data. This data is available from the IRG and CSENet as well. Note that IRG data provided by CSENet is in the same format as that provided on the IRG Web site at http://ocse3.acf.hhs.gov/ext/irg/sps/selectastate.cfm. (To gain access to the data, states must enter a username and password.)

3.2.5.1 Requesting IRG Data

An IRG Update file contains IRG records that have been updated since the last IRG download; the IRG Master file contains all FIPS codes and addresses on the IRG Web site. States may request the IRG Master file, the IRG Update file, or both, by providing appropriate data set names to the CSENet Service Desk. The Master file is normally used for the first IRG data load into a state system, whereas the Update file can be used to update the original data.

3.2.5.2 Defining Data Set Locations

There must be a designated location on the state system into which CSENet delivers IRG data. Depending on the state's operating system, the location of the IRG data will be either one or two mainframe data sets, or a directory and one or two filenames. One data set or file is needed for the IRG Master file and another is needed for the IRG Update file. (See Figure 3-6 on page 3-15 for examples of production data set names.) The name of the data sets or files in which to place the IRG data are unique to each state. The IRG data file locations should be in the same region or directory as other CSENet files.

3.2.5.3 Create vs. Append to IRG Data Sets

The IRG Master File data set is overwritten each time a file is uploaded so that it never contains duplicate IRG records. The IRG Update File data set is uploaded into using the same method (appending or overwriting) used to upload the state's other CSENet files. Appending to a data set ensures that the interface never overwrites data before the state system is able to process it.

3.2.5.4 Delivery of IRG Data to the State's CSE System

A state may obtain IRG data from CSENet through an automatic forwarding process. With this method, the state receives IRG data files from CSENet on the 15th and last business day of each month or on the last weekday before these days.

3.2.6 DISASTER RECOVERY SOFTWARE

The Disaster Recovery software provides CSENet continued operational support during critical outages through system redundancy. During a critical outage, the Disaster Recovery server functions as the operational Production server by taking over the Production server's tasks until it is operational again.

3.3 Software Management

The Software Enhancement and Change Process focuses on the management and quality assurance techniques used by OCSE in the FPLS Release Methodology found on the OCSE Web site. The Release Methodology promotes the following:

- software across all systems is released simultaneously;
- a schedule is published with standard release cycles;
- states are given an opportunity for planning and comment; and
- state involvement through teleconferences.

Under certain circumstances, emergency and out-of-cycle patches are required. When state systems are to be impacted by a change in the CSENet suite, state points-of-contact are informed in advance of the reason, type, and dates of the change and its impact to their systems.

With scheduled software releases, all changes follow the standard process of requirement and impact analysis. Once the software has completed the development and test integration phases, it is made available to a select group of states for additional or beta testing. After successful beta testing, the new software is released into production and made available to the CSENet user community.

3.3.1 BETA TESTING

All software enhancements and changes applied to CSENet undergo rigorous technical and functional in-house testing. Additionally, all software changes having an impact on state system programming are tested and evaluated with a group of volunteer states before being sent to the production system (pilot testing). If the change has no direct impact on states, it is thoroughly tested by the CSENet team before production release, but is not piloted.

3.3.2 PRODUCTION RELEASE

Software is released into the production environment only after a thorough and successful evaluation period on the CSENet test system. Once released, the performance of the new software is monitored for a period of time to verify the success of the implementation into production.